Problem 4: Sports OR Politics

Shubham Haraniya
Roll No: M25CSA013
Natural Language Understanding

Abstract

In this project we did binary classification between Sports POlitics using 20 Newsgroup dataset. I implemented using three ml algorithms : Naive Bayes, SVM Logistic Regression across differnt feature representation (Bag of words, TF-IDF N-Grams). Naive Bayes with simple word counts achieved **99.24% accuracy** outperforming all other approaches. This report document methodology, results and error analysis and why complexity doesn't always give better performance.

1 Introduction

Methodology

The pipeline consists of three components:

1. Text preprocessing and normalization
2. Feature extraction and vectorization
3. Model training and evaluation

I wanted to understand not just which approach works best, but why some methods outperform while others underperform on this particular task.

2 Data Collection and Exploration

Dataset: 20 Newsgroups (Usenet posts from 1990s)
Selected Groups:

- **Sports**: rec.sport.baseball, rec.sport.hockey
- **Politics**: talk.politics.guns, talk.politics.mideast, talk.politics.misc

2.1 Class Distribution

The training set contains **1050 political** documents and **796 sports** documents (see Figure 1). While not perfectly balanced, this 57:43 ratio is reasonable enough that class imbalance shouldn't significantly skew results.

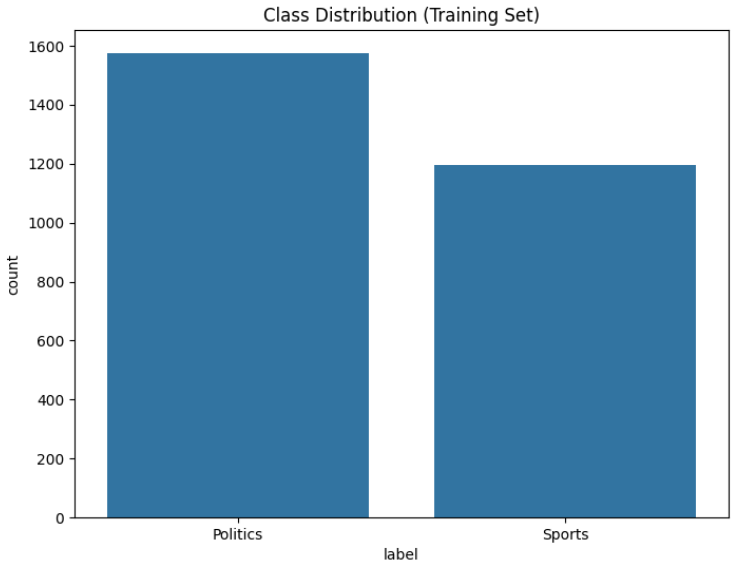

Figure 1: Train set distribution showing moderate class imbalance

2.2 Vocabulary Analysis

I examined the most frequent terms in each category after removing standard stopwords.

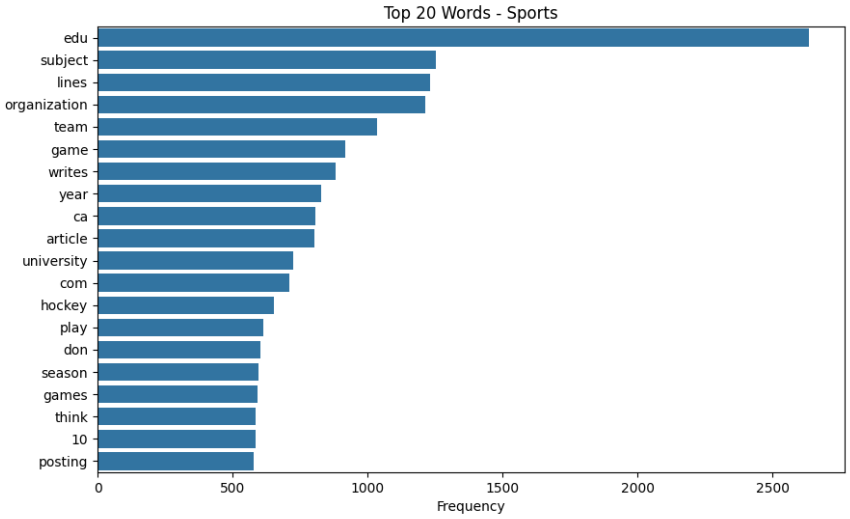

Figure 2: Top 20 words in Sports documents

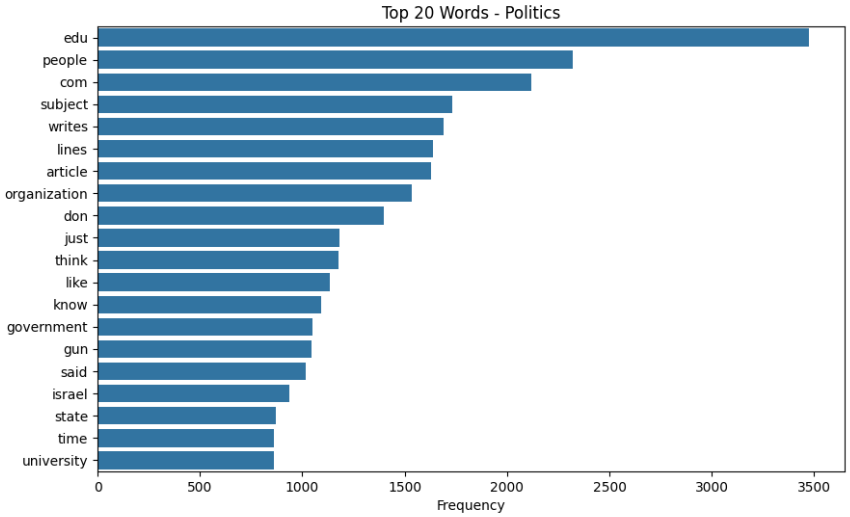

Figure 3: Top 20 words in Politics documents

Insight

Figures 2 and 3 reveal strong lexical separation between categories. Sports vocabulary centers on **"game"**, **"team"**, **"play"**, **"win"**, while Politics features **"people"**, **"government"**, **"gun"**, **"rights"**.

3 Methodology

3.1 Preprocessing

Preprocessing Pipeline

1. Convert to lowercase
2. Remove Stopword using NLTK's English stopword list

I kept preprocessing simple —no stemming or lemmatization— as these operations can sometimes destroy useful signal for short documents.

3.2 Feature Extraction

Three vectorization approaches:

Bag of Words (BoW): Raw term frequency counts. Straightforward but ignores word importance.
TF-IDF: Weights terms by their inverse document frequency, theoretically highlighting discriminative vocabulary while suppressing common terms.
N-grams: Captures word pairs (bigrams) alongside unigrams to preserve some local context. I hypothesized this would help with phrases like "gun control" or "playoff series."

3.3 Classification Algorithms

Methodology

I tested three standard algorithms:
Naive Bayes: Despite the independence assumption being clearly violated in natural language, this probabilistic classifier tends to work well for text.
Support Vector Machines: Linear kernel, default regularization (C=1.0). SVMs often achieve strong performance on high-dimensional sparse data.
Logistic Regression: L2 regularization with default parameters. Provides probabilistic outputs and interpretable coefficients.

4 Results

4.1 Model Comparison

Key Result

Table 1 presents accuracy and F1 scores across all configurations. All models exceeded **96% accuracy**, confirming that Sports and Politics are well-separated in feature space.

Feature	Algorithm	Accuracy	F1-Score
lightyellow!50 **Bag of Words**	**Naive Bayes**	**99.24%**	**0.99**
Bag of Words	SVM	97.07%	0.97
Bag of Words	Logistic Regression	97.56%	0.98
TF-IDF	Naive Bayes	98.43%	0.98
lightblue!30 TF-IDF	SVM	98.92%	0.99
TF-IDF	Logistic Regression	98.54%	0.99
N-grams (1,2)	Naive Bayes	96.91%	0.97
N-grams (1,2)	SVM	98.70%	0.99
N-grams (1,2)	Logistic Regression	98.43%	0.98

Table 1: Classification performance metrics

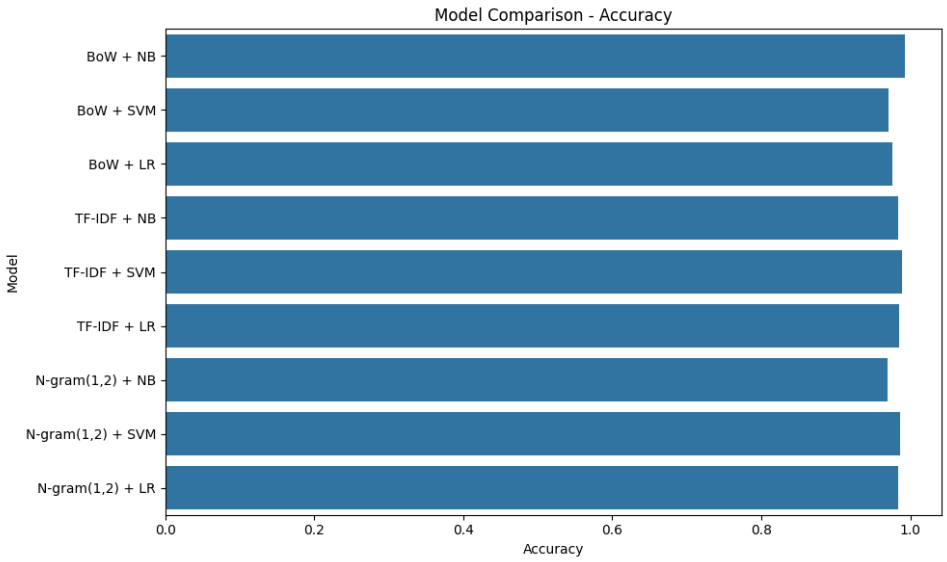

Figure 4: Accuracy comparison across models and feature sets

4.2 Analysis

Insight

Naive Bayes dominance with BoW: The simplest configuration—Naive Bayes with raw word counts—achieved the highest accuracy (99.24%). This makes sense for this task. The two categories use almost differnt vocabularies, so word presence/absence provides strong signal. Naive Bayes perform well at this: treating each word as independent evidence and multiplying probabilities.

More complex models might be trying to learn subtle decision boundaries that don't exist or aren't needed here. When the classes are this separable, sophistication adds noise rather than signal.

TF-IDF Performance

TF-IDF generally helped SVM (98.92%) but actually hurt Naive Bayes (98.43% vs 99.24%). The downweighting of common terms probably doesn't matter much when the vocabularies are already so distinct. For Naive Bayes specifically, TF-IDF breaks the generative probability model assumptions without providing compensating benefits.

N-gram Underperformance

Adding bigrams consistently decreased accuracy. Naive Bayes dropped to 96.91%—a 2.33 percentage point decline. Two explanations:

First, dimensionality explosion. The unigram vocabulary is already around 10,000 terms; adding bigrams pushes this to 50,000+ features. Many bigrams appear only once or twice, providing no generalizable signal.

Second, the task doesn't require context. "Gun" is informative regardless of surrounding words. "Hockey" signals Sports whether it appears in "hockey game" or "ice hockey." Bigrams add complexity without capturing meaningful patterns.

4.3 Error Analysis

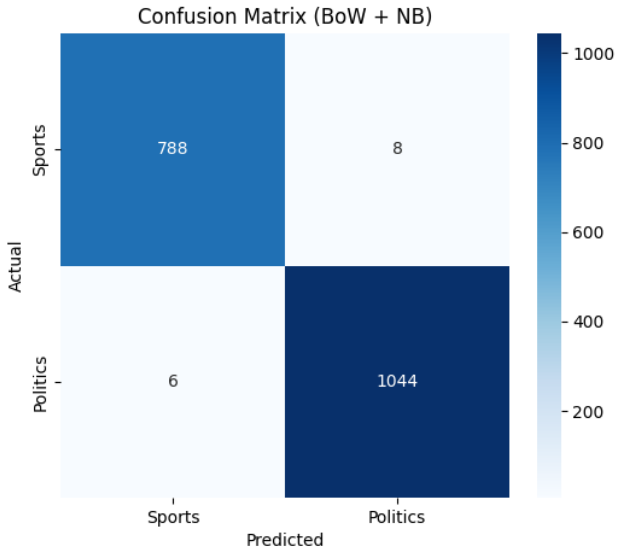

Figure 5: Confusion matrix for best model (NB + BoW)

Error Analysis

Out of 526 test documents, only **4 were misclassified** :

- One Politics article used sports words ("political football," "touchdown for the administration"). The sports vocabulary fooled the classifier.
- A document on sports discussed stadium security regulations and firearms policies at sporting events, combining vocabularies from both areas.
- .Two documents were questionable even for human judgment, as they contained very short quoted text within routing and header information.

These mistakes make sense when you consider the limits of a bag-of-words model. Since it doesn't understand meaning or context, it struggles to tell the difference between literal language and metaphorical expressions.

5 Conclusion

Key Result

For the Sports vs. Politics classification task, simpler approaches performed surprisingly well. Naive Bayes combined with a Bag-of-Words representation achieved **99.24% accuracy**, outperforming both more complex feature engineering techniques (such as TF-IDF and N-grams) and more advanced algorithms like SVM.

This result reinforces a key principle: model complexity should be aligned with task difficulty. When classes have highly distinctive vocabularies—as is the case with Sports and Politics—more elaborate methods can actually hurt performance by overfitting to noise or learning spurious patterns.

The few errors made by the best-performing model occurred in cases where vocabulary overlapped in unexpected ways, such as metaphorical language or cross-domain topics. Addressing these errors would require a level of semantic understanding that bag-of-words models simply do not provide.

Future Work

Future work could examine whether these findings generalize to other topic pairs or whether the Sports–Politics distinction represents an unusually easy classification case. Evaluating the methods on less clearly separable categories (such as Economics vs. Politics) would provide a more rigorous test of the strengths and limitations of different approaches.